Improving the flexibility of Active Grids through Web Services

Laurent Lefèvre
INRIA
LIP (UMR CNRS, INRIA, ENS, UCB)
École Normale Supérieure de Lyon
Lyon, France
laurent.lefevre@inria.fr

Paul Roe
Faculty of Information Technology
Queensland University of Technology
Brisbane, Australia
p.roe@qut.edu.au

Abstract

Active Grids are a form of grid infrastructure where the grid network is active and programmable. These grids directly support applications with value added services such as data migration, compression, adaptation and monitoring. Services such as these are particularly important for eResearch applications which by their very nature are performance critical and data intensive.

We propose an architecture for improving the flexibility of Active Grids through web services. These enable Active Grid services to be easily and flexibly configured, monitored and deployed from practically any platform or application. The architecture is called WeSPNI ("Web Services based on Programmable Networks Infrastructure"). We present the architecture together with some early experimental results on using web services to monitor data movement in an active grid. [1]

Keywords: active and programmable networks, grid computing, web services

1 Introduction

eResearch is an important new research area which uses Grid computing to virtualize people, resources and information. This enables the active and dynamic collaboration of researchers from different organizations and across different geographic regions. eResearch enables collaborative research. Grid computing provides the infrastructure for eResearch it enables the transparent sharing of information, computational resources, and devices.

eResearch applications running on grids are typically performance critical and data intensive. To address this active networking has been applied to Grids to yield Active Grids where the grid network is active and programmable. This enables applications and users to customize the grid for specific applications through value added services which can perform dedicated operations (for example, adapt and compress data, monitor data communications and migrate data).

Active Grids (Gelas & Lefèvre 2002) build on the Active Networking vision, routers can perform computations on user data in transit and users can modify the behavior of the network by supplying programs, called Programmable Network Services (PNS) that perform these computations. These routers are called active routers and provide great flexibility towards the deployment of new functionality, more adapted to architecture, users and service providers' requirements.

INRIA researchers have proposed the first execution environment Tamanoir(Gelas, El Hadri & Lefèvre 2003, Lefèvre 2005) designed for the support of Gigabit networks and to provide Grid network services. The Tamanoir Active node software suite is based on widely used components and tools: Myrinet (Boden, Cohen, Felderman, Kulawik, Seitz, Seizovic & Su 1995) and Giga Ethernet for NIC level, Netfilter/Linux for kernel space support, Java for user space level and Linux Virtual Server for distributed infrastructure. This Tamanoir software is deployed for experimental use on various local and long distance platforms.

Recent research has involved utilizing active network technology inside Grids in order to add the network component part to Grid infrastructure (Mohamed 2003). INRIA researchers have proposed solutions to merge the active networking and Grid fields by developing the Active Grid Architecture (A-Grid) which focuses on active network adaptation for supporting Grid environments and applications (Gelas & Lefèvre 2002). This active Grid Architecture allows solutions to support multi-cluster and P2P Grid configurations. In this architecture the network will take part in the Grid computing session by providing efficient and intelligent services dedicated to Grid data stream transport. This Active Grid provides an overlay infrastructure on Grids.

However until now managing PNS in Active Grids has been done through ad hoc protocols and services. We propose an architecture where services in Active Grids are managed through web services. This enables programmable network services to be easily and flexibly configured, monitored and deployed from practically any platform or application. Furthermore web services themselves can be used by Active Grid services to further enhance their processing capabilities, to provide non-performance critical functionality. While Web Services can now be used in various gateways and end host equipment (Bulut, Wu, Fox, Uyar, Pallickara & Altay 2004), mixing active networks and web services is still not deeply explored (Takashi, Minoru & Hideo 2000). Web services are important for next generation networking and grid computing; in particular the next generation Globus (the de facto standard grid computing Middleware (*Globus Grid Middleware* n.d.)) infrastructure is now adopting web services for communications. Web services are highly configurable and can be described by lots of semantic information.

In this paper we present the architecture and some

[1]This research is partially supported by FAST project number 10466QH between French Ministry of Foreign Affairs and Australian Government.

preliminary results for Active Grids utilizing web services for improved flexibility.

The paper is organized as follows. Section 2 briefly describes the Active Grid infrastructure. The WeSPNI architecture is presented in Section 3. An eResearch scenario using the proposed framework is described in Section 4. Experimental evaluation on local platforms are analyzed in Section 5. Section 6 concludes the paper and presents the on-going work.

2 Inside the Active Grid infrastructure

2.1 Programmable network nodes

We have designed an architecture for a high performance active router capable of being deployed around a high performance backbone: the Tamanoir Execution Environment (Gelas et al. 2003, Lefèvre 2005).

We define an Active Network Execution Environment (EE) as an environment able to *load* and deploy network services. It must be also able to *direct* packets towards the required service thanks to appropriate header filtering.

Programmable Network Services can be deployed at various levels depending on resources (e.g. processing capabilities, memory consumption and storage capacity) and intelligence (flexibility of the execution environment) they need. In order to provide an adapted EE for each type of service and to limit packets ascent, we design an active node architecture on four levels: Network Interface card (NIC), Kernel space, User space and distributed resources. (see Figure 1).

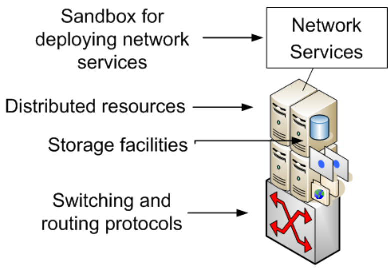

Figure 1: Tamanoir active node

2.2 Services deployment and repositories

There are three scenarios for active network service deployment (Figure 2) :

1. From end-nodes: The Grid Middleware or the Grid application provide network services to the programmable equipment (scenario *1* in figure 2).

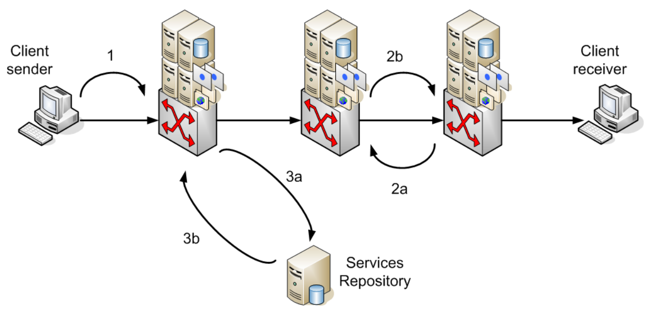

Figure 2: Programmable Network Services deployment

2. Node to node: the active node owning the requested service is able to provide the service to a requesting node. When a new service is requested, an active node is able to find it and request it to a node previously crossed in the data path (scenario *2* (*a*, *b*) in figure 2).

3. Services repository: specific servers (service repositories (see Figure 3)) are distributed in the Grid infrastructure. They own collections of active services and can provide them to requesting active nodes (scenario *3* (*a*, *b* in Figure 2).

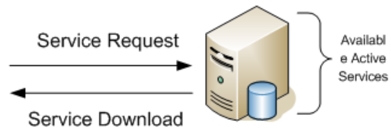

Figure 3: Services repository

2.3 Active Grid architecture

We propose an active network architecture dedicated to Grid environments and Grid applications requirements: the Active Grid architecture (Gelas & Lefèvre 2002).

An Active Grid architecture is based on a virtual topology of active network nodes spread on programmable routers of the network. Active routers, also called Active Nodes (AN), are deployed around backbone networks.

Since we are concerned with a wide active routers approach, we do not consider the deployment of Gigabit active routers in backbones. This is because the future of WAN backbones will be based on all-optical networks, no dynamic services will be allowed to process data packets. Thus active operations occur on edge routers/nodes mapped at network periphery.

Active nodes manage communications for a set of Grid nodes. Grid data streams cross various active nodes up to passive backbone and then cross another set of active nodes up to receiver node. The Active Grid architecture is based on Active Node approach: programs, called services, are injected into active nodes independently of data stream. Services are deployed on demand when streams arrive on an active node. Active nodes apply these services to process data streams packets.

To support most of Grid applications, the Active Grid architecture deals with the two main Grid configurations:

- Multi-cluster computing (Figure 4):

  In this highly coupled configuration, an active node is mapped on network head of each cluster or parallel machine. This node manages all data streams coming or leaving a cluster. All active nodes are linked with other AN mapped at backbone periphery. An Active node delivers data streams to each node of a cluster and can aggregate output streams to others clusters of the Grid.

- Global and P2P computing (Figure 4):

  In this loosely coupled configuration, an AN can be associated with each Grid node or can manage a set of aggregated Grid nodes. Hierarchies of active nodes can be deployed at each network heterogeneity point.

  Each AN manages all operations and data streams coming to Grid Nodes : subscribing operations of voluntary machines, results gathering, nodes synchronization and check-pointing. . .

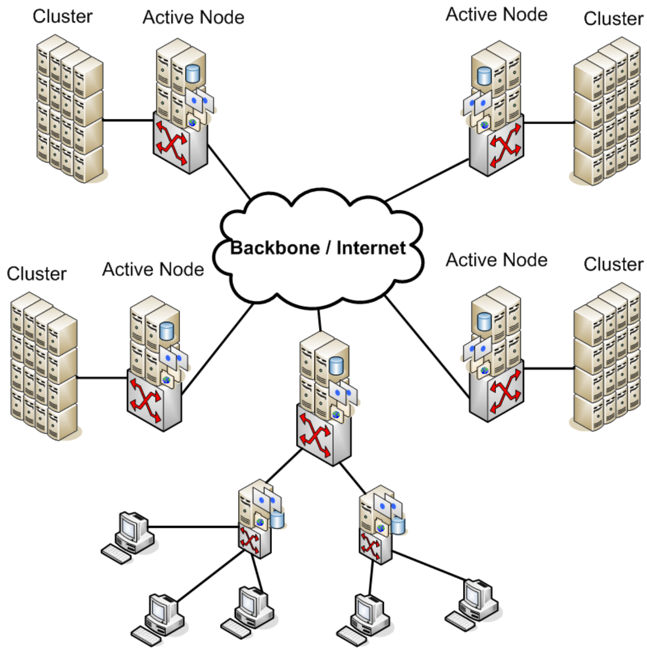

Figure 4: Multi-cluster and P2P computing Active Grid Architecture

For both configurations, active nodes will manage the Grid environment by deploying dedicated services adapted to Grid requirements, including: dynamic topology re-configuration, fault tolerance...

3 The WeSPNI architecture

We focus on the design of Web Services for Programmable Network Infrastructure (WeSPNI). We observe a real convergence between Grid infrastructure and Web Service solutions. Based on the Open Grid Service Infrastructure, Grid researchers have proposed the WSRF (Web Service Resource Framework) (Czajkowski, Ferguson, Foster, Frey, Graham, Sedukhin, Snelling, Tuecke & Vambenepe 2004) where Web Services support the stateful computation necessary for Grid computing. This convergence currently needs to be efficiently exploited by providing network solutions adapted to Grid requirements.

3.1 Flexible network services

Programmable network nodes must embed heterogeneous dynamic services from management and control plane (e.g. monitoring, accounting) to data plane (e.g. streams adaptation, storage on the fly) (see Figure 5).

In the active network research field, network services are pieces of code deployed on programmable network equipment. We propose to add the possibility of high level service composition through calls of web services from programmable network services.

In the WeSPNI architecture, network services consist of portions of local code combined with web services invocations (See Fig. 6). Active nodes support a pipelined execution of functionalities of PNS. This original approach can allow a flexible autonomic composition of services and can propose high level interactions between active nodes and migration of expensive functionality to remote servers.

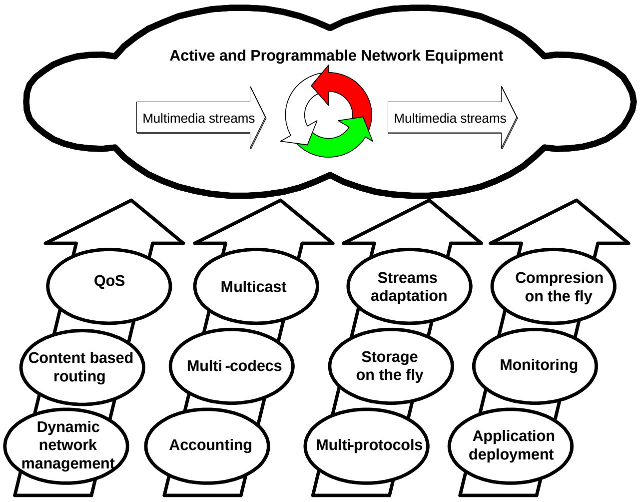

Figure 5: Classes of services

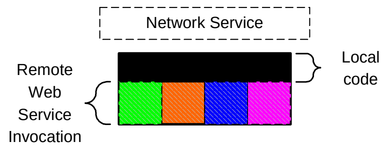

Figure 6: A network service

3.2 Management and Control plane

We propose an architecture where services in Active Grids are managed through web services. This enables services to be easily and flexibly configured, monitored and deployed from practically any platform or application.

With the WS-Management (Arora, Geller, He, Kaler, McCollum, Milenkovic, Montgomery, Saiyed & Suen 2005) framework, Web Services allow a flexible management of servers and remote devices. With this framework, active networks can be remotely managed and monitored in order to control the autonomic deployment of programmable network services.

3.3 Active networks and Web services for the data plane

- Deploying web services blocks inside networks: Currently available web services frameworks and implementation are based on generic and complex solutions. Lack of efficiency of these solutions remain an open problem. We propose to deploy some parts of Web Services inside dynamic network equipment by linking with programmable network solutions (Figure 7). This will allow efficient and flexible solutions for Web Services infrastructure. For example, in figure 7, a programmable network service is based on some code and four remote functions calls. These functions are available through web services in some network equipment. So the code is dynamically deployed on network nodes and benefits from Web Services on the nodes.

- Value added services inside non re-programmable networks: Deploying functionality inside networking equipment requires the dynamic programming of equipment. This is not always possible, thus WeSPNI explores solutions based on adaptive routing functionality inside the network, allowing the possibility to route streams to required services present on end machines.

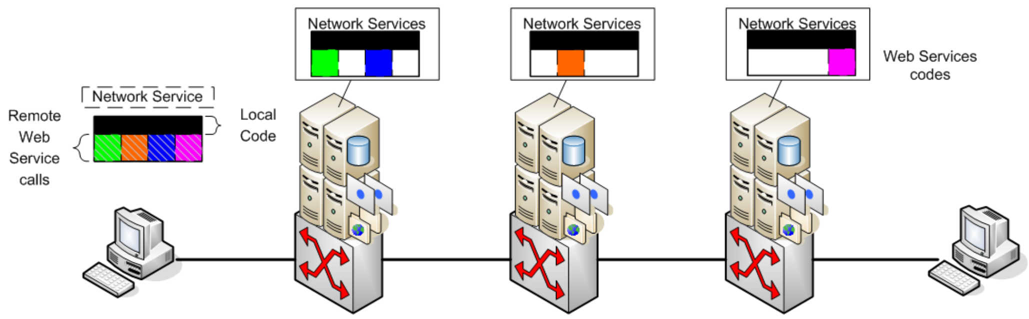

Figure 7: If network nodes are dynamically programmable equipments, some parts of Web Services are deployed inside the network.

In figure 8, network nodes have limited processing capabilities. So local code of programmable network services are deployed on active nodes and remote functions remain on distributed web servers off data path.

3.4 Exposing network services

To enable eResearch, end-users should be able to exchange and use dynamic network services provided by others Grid sites. So after deploying some network services, users should be able to advertise capabilities and functionality of these services. Web services are distributed application components that conform to standards that make them externally available. As active network services need to be deployed on heterogeneous platforms and called and interfaced with E2E applications, they face the same problem as web services.

Exposing an active network services means publishing a description of the service interface, as Web Services description consists of the service's Web Services Description Language (WSDL) description and XML schema referenced by the service description.

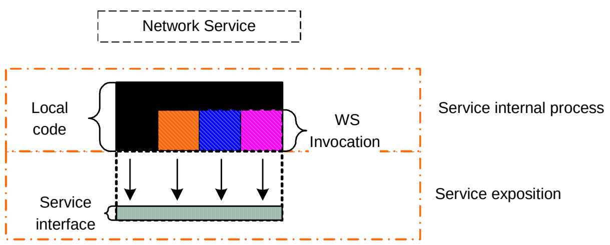

Figure 9: Exposing active network services capabilities

We propose a simplified exposition facility based on the triplet { *input, processing, output* }

- *Input*: programmable network services are packet based. So input parameters are fitted in data packets crossing the active nodes and can concern packet header or payload.
  input = { *packet header* || *packet payload* }
  These input parameters contain the service reference which must be deployed into the nodes and some specific states.

- *Processing*: contains the description of the functionality applied in programmable network services. It is based on a pipelined execution of local functions and remote functions through web services available on remote capabilities (see Figure 9).
  $processing = \{ service_1 \ \& \ service_i \ \& \ service_n \}$
  The description of the processing can also include the storage of some states useful for the service.

- *Output*: precises the level of modification from the PNS : discarding of a packet, duplicating broadcasting, modifying header or payload.
  output = { *modified header* || *modified payload* }

4 eResearch Scenario

In this section we describe a eResearch scenario requiring Active Networking, and argue how that is made more flexible through web services.

In eResearch, distributed sources (repositories) of data are linked by a Grid infrastructure. These data repositories are used for global collaborative computing purposes and are shared and moved between Grid sites to facilitate collaboration. In order to optimize data locality, the network should provide monitoring of applications level data communications through programmable network services. These value-added services (which must not be included in the application) can be enabled by the grid's active network nodes. The active nodes must exchange collected data between themselves and provide summaries of logging information to distributed portals (Figure 10). Users can dynamically change the frequency of monitoring collection to efficiently deal with the impact of intrusive collection on the overall performance. More generally higher level grid services may wish to automatically move and replicate data based on information gathered from these programmable network services, or to cache such data. Thus exposing PNS in a general way has many benefits and greatly adds to the flexibility of the active grid. Web services do exactly this. They enable the flexible exposure of programmable network services in a decoupled manner so that other platforms and applications can easily consume such information with no knowledge and hence dependency of the underlying active nodes implementation. The use of extensive meta-data in web services and standards for securing information further benefits the active grid.

5 Experimental Validation

The experimental platform consists of two Grid clients exchanging data streams through a Tamanoir active network node (TAN). This programmable node embeds dynamic services which are deployed from a service repository and which call remote web services (Figure 11).

We design the *MonitorS* active service as:

- a Java part deployed in the Tamanoir active nodes. This service monitors packets crossing the node and collect logging information. With a frequency specified by the end-users, the TAN compresses *logs* tables and exchanges it with remote active nodes or Web Servers.

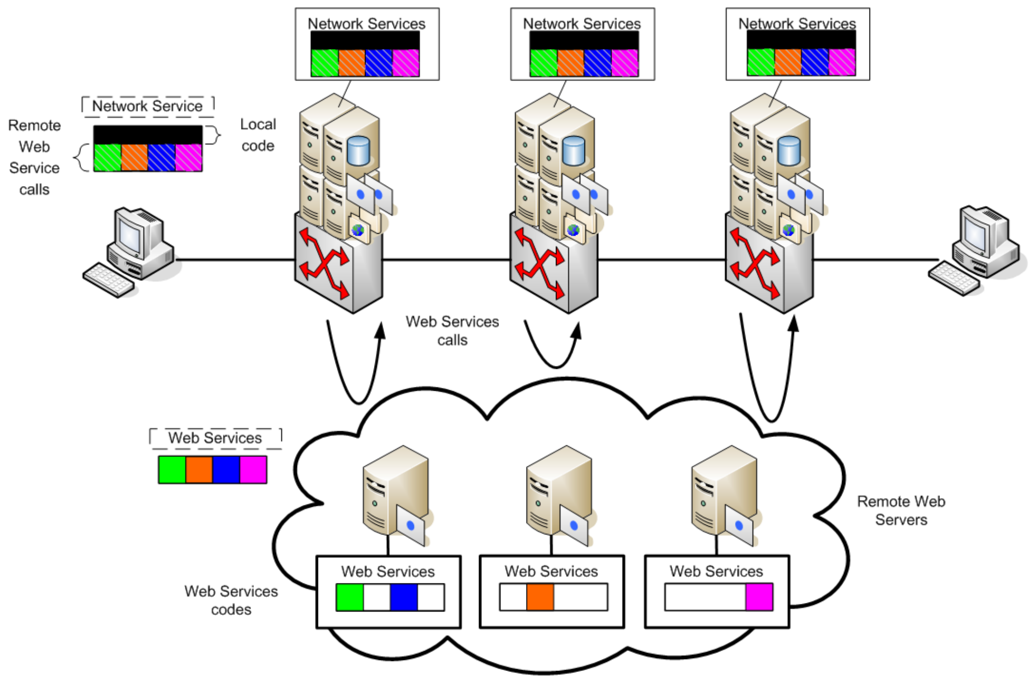

Figure 8: Web Services are deployed on Web Servers while adaptive routing is allowed inside network equipments

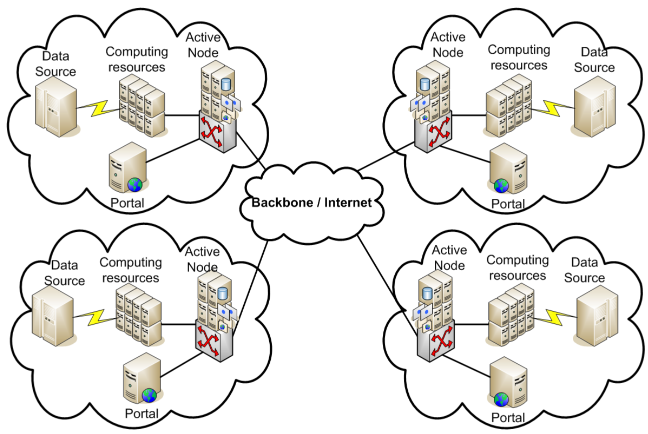

Figure 10: eResearch scenario

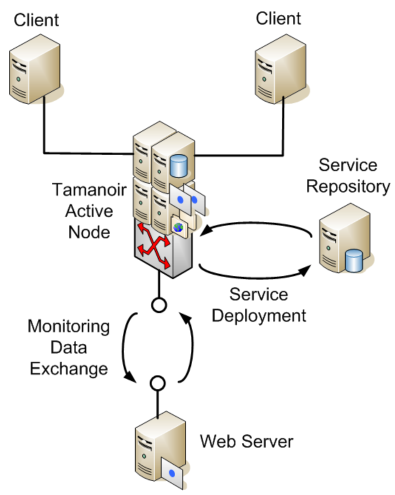

Figure 11: Experimental platform

- a Web Service part deployed in the Web Server (Tomcat / Axis). This web service collects logs from different TANs and provides information to distributed portals.

We evaluate the impact of a remote invocation to the web service due by modifying the frequency of update of logging table and show that the cost of sending uncompressed data to the Web Services through SOAP greatly impacts the performance of a Tamanoir Monitoring service, whose bandwidth is around 220 Mbits (see figure 12). We show that compression of the logging table and increasing the size of data packets greatly improves the performance of the Web Service (see figure 13). Figure 13 also demonstrates the approximate bandwidth of the active node using a basic forwarding service.

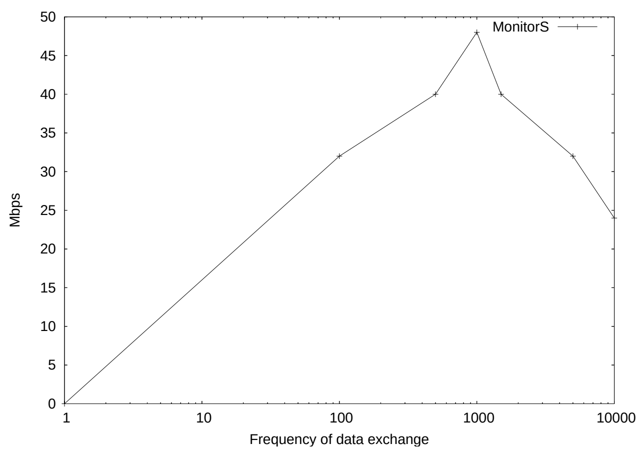

Figure 12: Cost of communications with web services from active service

6 Conclusion and future works

In this paper we have proposed using web services to make the active grid architecture more flexible. Web services enable services of active nodes to be monitored, managed, deployed and controlled easily. The services for doing this are well described

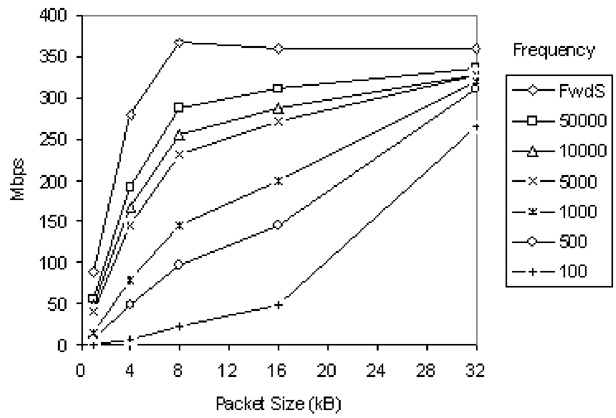

Figure 13: Communication performance improvements due to compression

by web service metadata and can leverage other web service infrastructure, such as security, as necessary. We have undertaken some initial experiments using web services to provide a monitoring interface to active nodes. The example interface enables application level data communication to be monitored; this can be used for a variety of purposes including monitoring portals or data oriented middleware. These first experiments have used HTTP and Unicode based web services, the performance of which limits the use to non-performance critical code, off the main data path. Our long term goal is to move remote functionality to distributed in path and out of the path infrastructure. We believe binary implementations of web services will make this feasible.

Acknowledgments

The authors wish to thank Chien Jon Soon for his help during the experimental validation step.

References

Arora, A., Geller, A., He, J., Kaler, C., McCollum, R., Milenkovic, M., Montgomery, P., Saiyed, J. & Suen, E. (2005), 'Web services for management'.

Boden, N., Cohen, D., Felderman, R., Kulawik, A., Seitz, C., Seizovic, J. & Su, W.-K. (1995), 'Myrinet : a gigabit per second local arear network', *EEE-Micro* .

Bulut, H., Wu, W., Fox, G., Uyar, A., Pallickara, S. & Altay, H. (2004), A web services based streaming gateway for heterogeneous a/v collaboration., *in* 'International Conference on Internet Computing', pp. 493–502.

Czajkowski, K., Ferguson, D., Foster, I., Frey, J., Graham, S., Sedukhin, I., Snelling, D., Tuecke, S. & Vambenepe, W. (2004), 'The WS-Resource Framework'. http://www.globus.org/wsrf/.

Gelas, J.-P., El Hadri, S. & Lefèvre, L. (2003), 'Towards the design of an high performance active node', *Parallel Processing Letters journal* **13**(2).

Gelas, J.-P. & Lefèvre, L. (2002), Towards the design of an active grid, *in* L. N. in Computer Science, ed., 'Computational Science - ICCS 2002', Vol. 2230, Amsterdam, The Netherlands, pp. 578–587. ISBN 3-540-43593-X.

Globus Grid Middleware (n.d.). http://www.globus.org.

Lefèvre, L. (2005), Heavy and lightweight dynamic network services : challenges and experiments for designing intelligent solutions in evolvable next generation networks, *in* I. Society, ed., 'Workshop on Autonomic Communication for Evolvable Next Generation Networks - The 7th International Symposium on Autonomous Decentralized Systems', Chengdu, Jiuzhaigou, China, pp. 738–743.

Mohamed, N. (2003), Active networks and their utilization in the computational grid. Technical Report TR03-04-01, Computer Science and Engineering, University of Nebraska-Lincoln.

Takashi, N., Minoru, K. & Hideo, O. (2000), Large-scale high-quality communication service solution using active network technology, *in* 'Hitachi Review', Vol. 49.